Sharif University of Technology

ML project

Phase 1

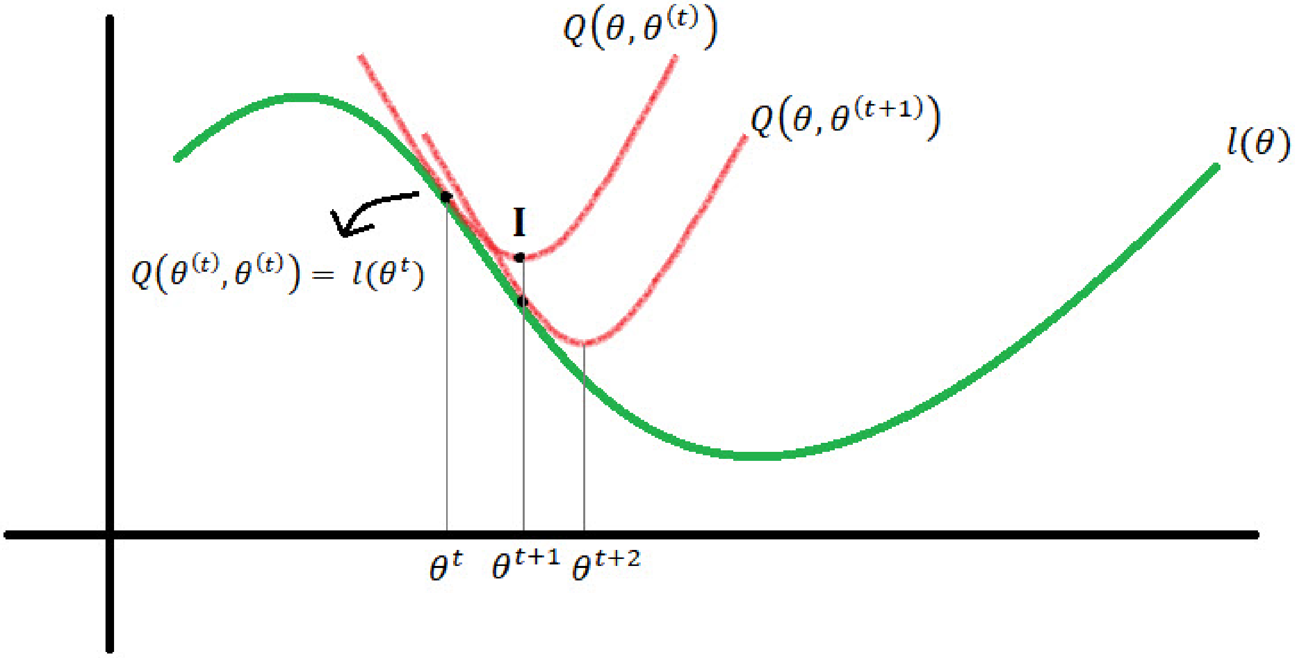
Ardalan Gerami

99102112

Theory Questions

Theory Question1.

consider the graph :


$Q(\theta, \theta^{(t)})$
$Q(\theta, \theta^{(t+1)})$
$l(\theta)$
I
$Q(\theta^{(t)}, \theta^{(t)}) = l(\theta^t)$
θ^t
θ^{t+1}
θ^{t+2}

We know the MM algorithm works by finding a surrogate function that minorize or majorize the objective function.

Optimizing $l(\theta)$ is not easy so we defined surrogate function.

as we can see $l(\theta)$ is a non-convex optimization objective function. $Q(\theta)$ is a simple convex surrogate function which has some special properties due to $l(\theta)$, $Q(\theta, \theta^{(t)})$ is a tight upper bound to $l(\theta)$ such that :

$$Q\left(\theta^{(t)}, \theta^{(t)}\right) = \ l(\theta)$$

$$Q\left(\theta, \theta^{(t)}\right) \geq \ l(\theta)$$

* as we see in graph point (I) shows our majorize.

The next step we check the MM algorithm is $\theta^{(t+1)}$ which define as:

$$\theta^{(t+1)} = \arg\min_{\theta} Q\left(\theta, \theta^{(t)}\right)$$

This gaurantees us monotic $Q(\theta)$ monotically decrease in $l(\theta)$ and in conclusion:

$$l(\theta^{(t+1)}) \leq Q\left(\theta^{(t+1)}, \theta^{(t)}\right) \leq Q\left(\theta^{(t)}, \theta^{(t)}\right) \leq \ l(\theta)$$

Theory Question2.

As we see in formula

$$P(y;\theta) \ = \sum_{k=1}^{K} P(y, Z \ = \ z_k; \ \theta)$$

Obviously Calculating $P(y;\theta)$ is hard, define a new vector Z which contains latent variables.

Rewrite:

$$P(y;\theta) = \sum_{k=1}^{K} P(z_k; \ \theta)P(y|Z \ = \ z_k; \ \theta)$$

Now we have distribution of Z, due to formula we can optimize $P_{Y,Z}(y_n, z_n; \theta)$ easier.

Theory Question4.

1.

Above all determine model parameters and initialize them :

Πk → the prior probability of the kth GD.

μk → the mean vector of the kth GD.

Σk → the covariance matrix of the kth GD.

2.

according to definition we know:

$$q_{i,k} = p\left(z^{(i)} = k \middle| X^{(i)}\right) = \frac{p\left(z^{(i)} = k\right)p(X^{(i)}|z^{(i)} = k)}{\sum_{j=1}^{k} p(z^{(i)} = j)p(X^{(i)}|z^{(i)} = j)}$$

$$= \frac{\pi_k {}_{N(}X^{(i)}; \mu_k, \Sigma_k)}{\sum_{j=1}^{k} \pi_j {}_{N(}X^{(i)}; \mu_j, \Sigma_j)}$$

And

$$p(D;\theta) = \prod_{i=1}^{N}\sum_{k=1}^{K} p(X_i|z_i = k;\theta)p(z_i = k) = \prod_{i=1}^{N}\sum_{k=1}^{K} q_{i,k}\frac{\pi_k N(X^{(i)};\mu_k,\Sigma_k)}{q_{i,k}}$$

Which $p(D;\theta)$ is complete dataset likelihood.

3.

E-step :

$$q_{i,k}^{t} = \frac{\pi_k^t N\left(X^{(i)};\mu_k^t,\Sigma_k^t\right)}{\sum_{j=1}^{k}\pi_j^t N\left(X^{(i)};\mu_j^t,\Sigma_j^t\right)}$$

M-step:

We know optimization of $l = \log N(X;\mu_k,\Sigma_\mathrm{k})$:

$$N(X;\mu_k,\Sigma_\mathrm{k}) : (2\pi)^{-\frac{K}{2}}\det(\Sigma)^{-\frac{1}{2}}\exp(-\frac{1}{2}(X-\mu)^T\Sigma^{-1}(X-\mu)$$

$$\frac{\partial l}{\partial\mu_k} = 0 \rightarrow \mu_k = \frac{\sum_{n=1}^{N}X^n}{N}$$

$$\frac{\partial l}{\partial\Sigma^{-1}} = 0 \rightarrow \Sigma_\mathrm{k} = \frac{\sum_{n=1}^{N}(X^n-\mu_k)(X^n-\mu_k)^T}{N}$$

Now if we generalize these conclusions to optimization of $Q\left(\theta,\theta^{(t)}\right)$:

$$\pi_k^{t+1} = \frac{\sum_{i=1}^{N}q_{i,k}^t}{N}$$

$$\mu_k^{t+1} = \frac{\sum_{i=1}^{N}q_{i,k}^t X^{(i)}}{\sum_{i=1}^{N}q_{i,k}^t}$$

$$\Sigma_k^{t+1} = \frac{\sum_{i=1}^{N}q_{i,k}^t\,(X^{(i)}-\mu_k)(X^{(i)}-\mu_k)^T}{\sum_{i=1}^{N}q_{i,k}^t}$$

Theory Question5.

1.

Above all determine model parameters and initialize them :

Π_k → the prior probability of the kth CD.

θ_k → the PMF vector of the kth CD.

2.

according to definition we know:

$$q_{i,k} = p\left(z^{(i)} = k \middle| X^{(i)}\right) = \frac{p\left(z^{(i)} = k\right)p(X^{(i)}|z^{(i)} = k)}{\sum_{j=1}^{k} p(z^{(i)} = j)p(X^{(i)}|z^{(i)} = j)}$$

$$= \frac{\pi_k Cat(X^{(i)}; \theta_k)}{\sum_{j=1}^{k} \pi_j Cat(X^{(i)}; \theta_k)}$$

And

$$p(D;\theta) = \prod_{i=1}^{N} \sum_{k=1}^{K} p(X_i|z_i = k; \theta)p(z_i = k) = \prod_{i=1}^{N} \sum_{k=1}^{K} q_{i,k} \frac{\pi_k Cat(X^{(i)}; \theta_k)}{q_{i,k}}$$

Which $p(D;\theta)$ is complete dataset likelihood.

3.

E-step :

$$q_{i,k}^{t} = \frac{\pi_k^t Cat\left(X^{(i)}; \theta_k^t\right)}{\sum_{j=1}^{k} \pi_j^t Cat\left(X^{(i)}; \theta_j^t\right)}$$

M-step:

We know optimization of $l = \log N(X; \mu_k, \Sigma_{\mathrm{k}})$:

$$Cat(X; \theta_k) : \prod_{j=1}^{K} \theta_j^{[x=j]}$$

$$\theta_k = \frac{\sum_{n=1}^{N} X^n}{N}$$

Now if we generalize these conclusions to optimization of $Q\left(\theta, \theta^{(t)}\right)$:

$$\pi_k^{t+1} = \frac{\sum_{i=1}^{N} q_{i,k}^{t}}{N}$$

$$\theta_k^{t+1} = \frac{\sum_{i=1}^{N} q_{i,k}^{t} X^{(i)}}{\sum_{i=1}^{N} q_{i,k}^{t}}$$

Simulation Question

Above all we explain the algorithm :

Here we have used Euclidan distance.

First consider a random mean and variance matrix for 3 distirbutions.

Now in E-step function we calculate the distance between mean of 3 distributions and data. every data that has the least distance from mean will be in that cluster and in matrix R the i^{th}data which is assigned to j^{th} cluster will get value 1.

After clustering datas, then we calculate new mean and variance matrix of each cluster(distribution)and again repeat the algorithm with new initial variables. until data converges to logical values.

Simulation Question1.

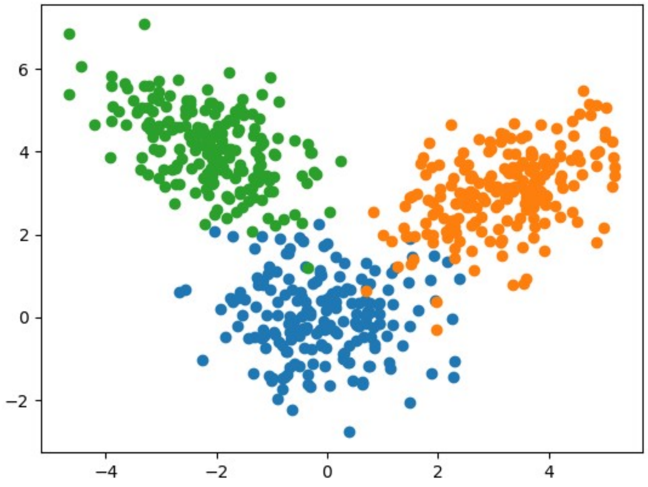


Image1

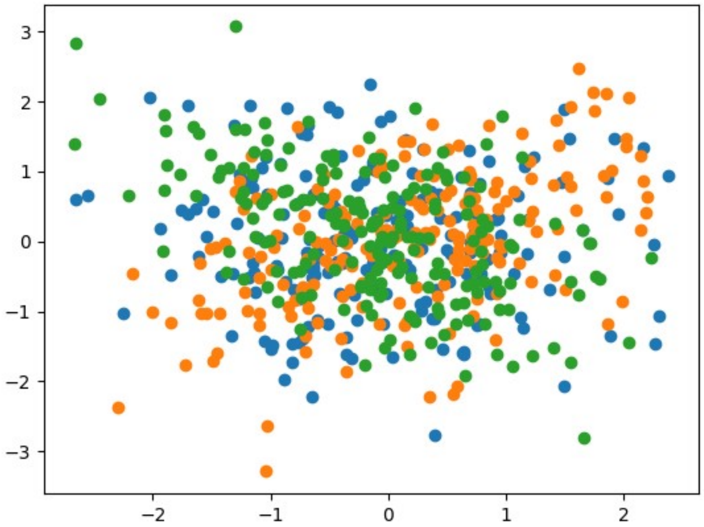
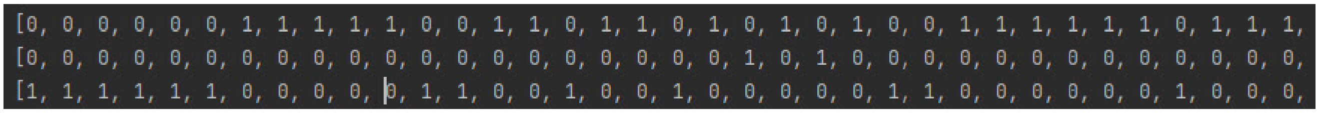


Image2

Simulation Question2.

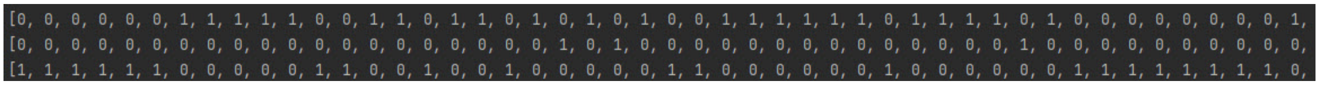
R_{ij} for some distibutions(one iteration):



```
[0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 0, 0, 1, 1, 0, 1, 1, 0, 1, 0, 1, 0, 1, 0, 0, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1,
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
[1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0,
```

Image1

R_{ij} for some distibutions(one iteration):

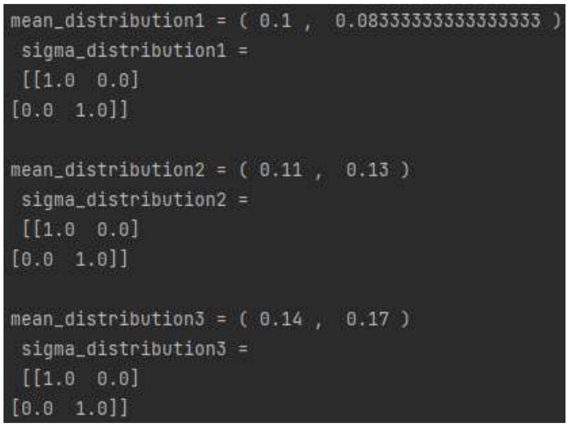


```
[0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 0, 0, 1, 1, 0, 1, 1, 0, 1, 0, 1, 0, 1, 0, 0, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1,
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
[1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 0,
```

Image2

Simulation Question3.

Initial values:

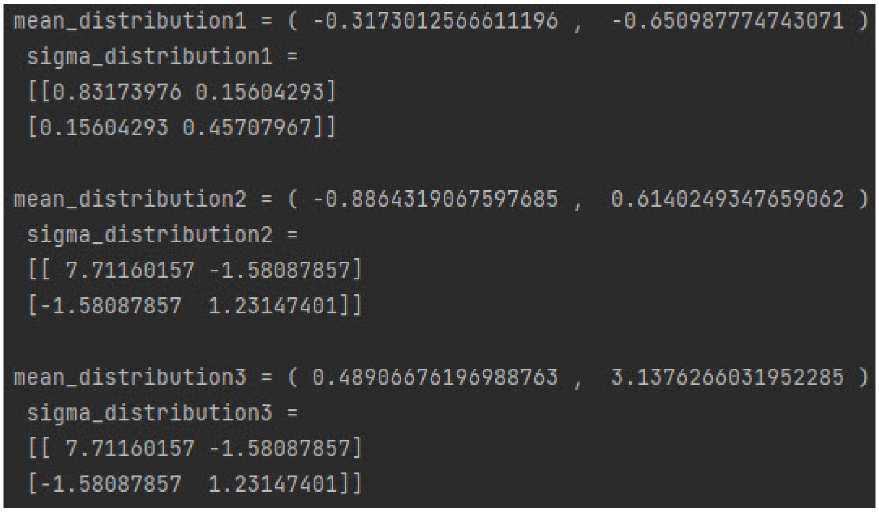


```
mean_distribution1 = ( 0.1 ,  0.08333333333333333 )
 sigma_distribution1 =
 [[1.0  0.0]
[0.0  1.0]]

mean_distribution2 = ( 0.11 ,  0.13 )
 sigma_distribution2 =
 [[1.0  0.0]
[0.0  1.0]]

mean_distribution3 = ( 0.14 ,  0.17 )
 sigma_distribution3 =
 [[1.0  0.0]
[0.0  1.0]]
```

After Run this function(E-step) for one iteration, new variances and means of each distribution:



```
mean_distribution1 = ( -0.3173012566611196 ,  -0.650987774743071 )
 sigma_distribution1 =
 [[0.83173976 0.15604293]
 [0.15604293 0.45707967]]

mean_distribution2 = ( -0.8864319067597685 ,  0.6140249347659062 )
 sigma_distribution2 =
 [[ 7.71160157 -1.58087857]
 [-1.58087857  1.23147401]]

mean_distribution3 = ( 0.48906676196988763 ,  3.1376266031952285 )
 sigma_distribution3 =
 [[ 7.71160157 -1.58087857]
 [-1.58087857  1.23147401]]
```

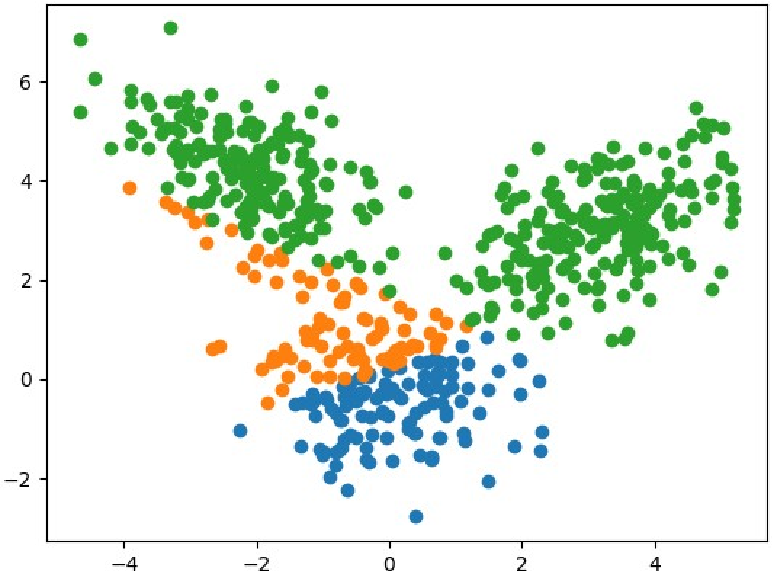
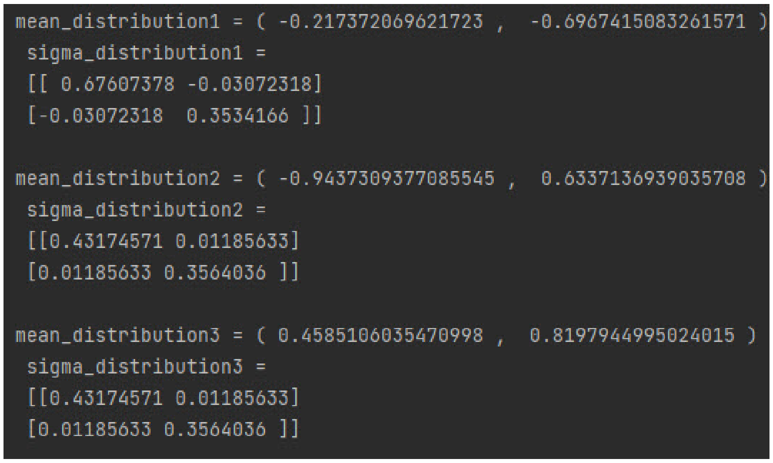


Image1



```
mean_distribution1 = ( -0.217372069621723 ,  -0.6967415083261571 )
 sigma_distribution1 =
 [[ 0.67607378 -0.03072318]
 [-0.03072318  0.3534166 ]]

mean_distribution2 = ( -0.9437309377085545 ,  0.6337136939035708 )
 sigma_distribution2 =
 [[0.43174571 0.01185633]
 [0.01185633 0.3564036 ]]

mean_distribution3 = ( 0.4585106035470998 ,  0.8197944995024015 )
 sigma_distribution3 =
 [[0.43174571 0.01185633]
 [0.01185633 0.3564036 ]]
```

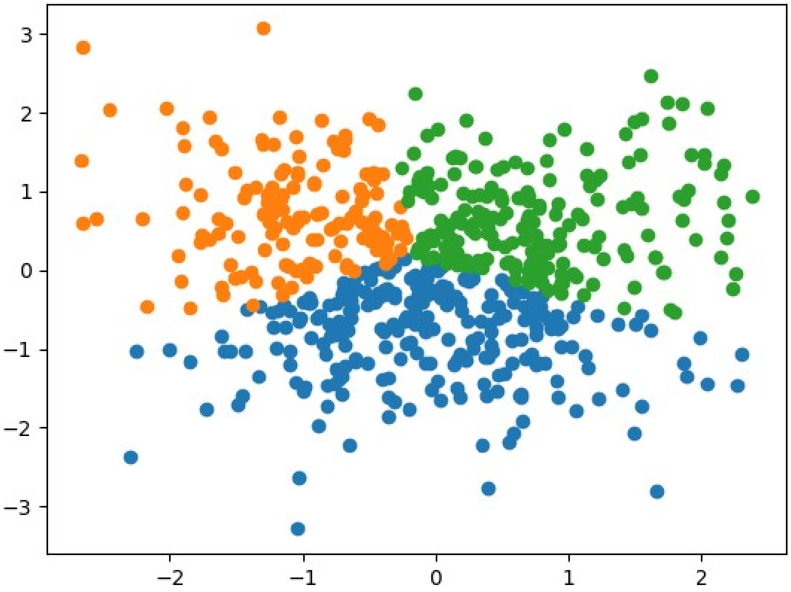
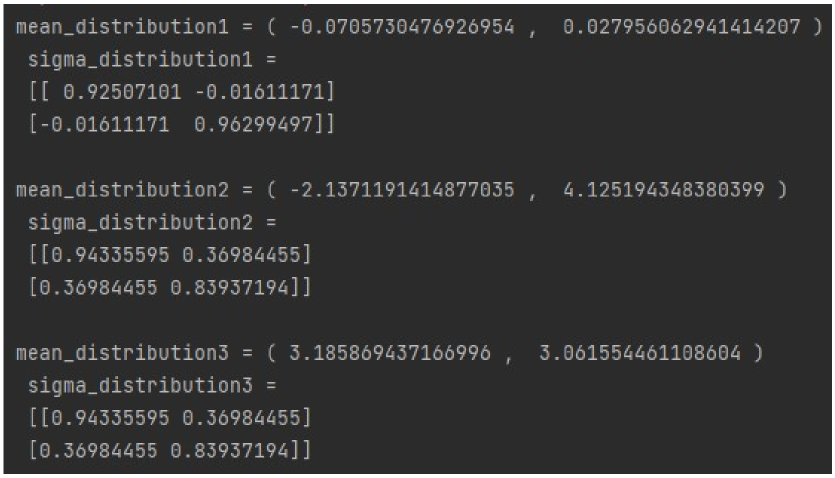


Image2

Simulation Question4.



```
mean_distribution1 = ( -0.0705730476926954 ,  0.027956062941414207 )
 sigma_distribution1 =
 [[ 0.92507101 -0.01611171]
 [-0.01611171  0.96299497]]

mean_distribution2 = ( -2.1371191414877035 ,  4.125194348380399 )
 sigma_distribution2 =
 [[0.94335595 0.36984455]
 [0.36984455 0.83937194]]

mean_distribution3 = ( 3.185869437166996 ,  3.061554461108604 )
 sigma_distribution3 =
 [[0.94335595 0.36984455]
 [0.36984455 0.83937194]]
```

.

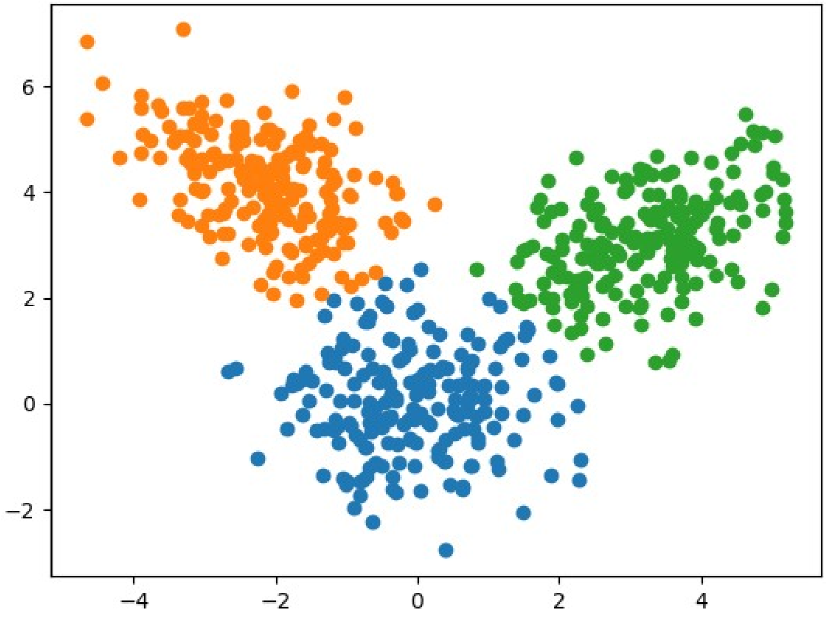
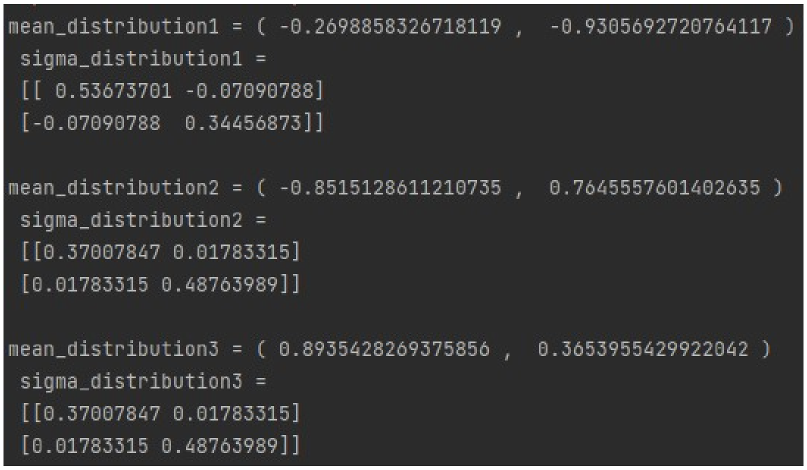


Image1



```
mean_distribution1 = ( -0.2698858326718119 ,  -0.9305692720764117 )
 sigma_distribution1 =
 [[ 0.53673701 -0.07090788]
 [-0.07090788  0.34456873]]

mean_distribution2 = ( -0.8515128611210735 ,  0.7645557601402635 )
 sigma_distribution2 =
 [[0.37007847 0.01783315]
 [0.01783315 0.48763989]]

mean_distribution3 = ( 0.8935428269375856 ,  0.3653955429922042 )
 sigma_distribution3 =
 [[0.37007847 0.01783315]
 [0.01783315 0.48763989]]
```

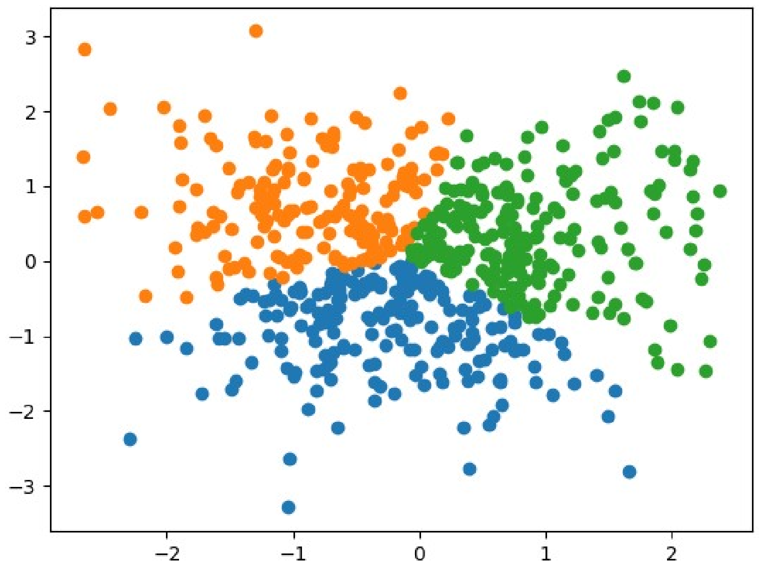


Image2

After convergence,

In image 1:

According to Q1 obviously we can cluster datas in 3 part because of different'means between datas.

And here after converging mean and variance we obtained almost the same clustering.

Also convergence is almost fast.

In image 2:

According to Q1 obviously we can't cluster datas in 3 part because mean of clusters have little distance so it is hard to devied them to 3 clusters and here convergence is slow.

Simulation Question5.

After run the EM algorithm the parameters of Image1&2 distributions are different from each other in 1 iteration until its convergence which is not suprising because ,as we mentioned in Q4, mean of Image1 distributions have enough distance from together which means clustering and convergence is easy and fast but Image2 distributions are centered and clustering is not as easy es Image1.